

Survey and Analysis of Web Service Composition Strategies: A State of Art Performance Study

J. Amudhavel^{1*}, U. Prabu¹, P. Inbavalli¹, N. Moganarangan², V. Ravishankar³,
R. Baskaran⁴ and P. Dhavachelvan⁵

¹Department of CSE, SMVEC, Pondicherry - 605107, Tamil Nadu, India;
info.amudhavel@gmail.com, uprabu28@gmail.com

²Department of CSE, RGCT, Pondicherry - 607403, Tamil Nadu, India; inbacse90@gmail.com

³Department of CSE, Bharathiar University, Coimbatore - 641046, Tamil Nadu, India; rengannsj77@gmail.com

⁴Department of CSE, Anna University, Chennai - 600025, Tamil Nadu, India; baaski@annauniv.edu

⁵Department of CSE, Pondicherry University, Pondicherry - 605014, Tamil Nadu, India; dhavachelvan@gmail.com

Abstract

Background/Objectives: To analyze the functions and merits of web service composition and provide services to many number of users at a time. **Methods/Statistical Analysis:** The standard-based approach which used to connect the web service to create the business processes which are of higher level. There are various approaches for composing the web services. Web service integrates various web applications by using languages namely WSDL, XML, WSFL etc. **Findings:** In this research, Web Service Composition (WSC) aims to organize numerous services to fulfill the user needs. It is one of the flexible way of integration of applications and resource sharing. The algorithm used by web service composition are Evolutionary algorithm, Ant Colony Optimization, Genetic algorithm, Particle Swarm algorithm, Branch and Bound for Execution Plan Selection (BB4EPS) algorithm, Join-Idle-Queue (JIQ) algorithm and Two-phase algorithm. Through the web service leads to provide services to more than single user. **Applications/Improvements:** When solving the web service composition problem, composing a sequence of web services is converted into finding a chain of actions. Automated planning technique uses two steps for web service composition dynamically. In this approach, WSC problem is taken as input and translated into WSC planning problem. Web service composition explore their functions through the online process and business needs.

Keywords: Web Service Composition - Approaches

1. Introduction

Web service is a promising technology which achieves interoperability among various heterogeneous application. Web services are of two classes namely REST-complaint web services and Arbitrary web services. REST-complaint web services focuses on the stateless operations. Arbitrary web services exposes the arbitrary operations.

Many organizations use various types of the software for their managerial activities. For these activities to be carried on there is need for exchange of data. These data exchange is carried out by a communication named web service. A service requestor is the one which requests

for the data. It associates with the UDDI (Uniform Description, Discovery and integration) for finding the service provider. It uses the SOAP protocol to contact the service provider. A service provider processes the request and provides the data to service requestor. It provides data in the form of an XML file. The service which receiving the file validates it through the XSD file.

Web service composition aims to organize numerous services to fulfill the user needs. It is one of the flexible way of integration of applications and resource sharing. The algorithm used by web service composition are Evolutionary algorithm¹, Ant Colony Optimization², Genetic algorithm³, Particle Swarm algorithm, Branch⁴

* Author for correspondence

and Bound for Execution Plan Selection (BB4EPS) algorithm⁵, Join-Idle-Queue (JIQ) algorithm⁶, Two-phase algorithm⁷.

The QoS-Aware web service composition⁸ is categorized into two types namely static approach and adaptive approach. The static approach is sub-categorized into approximation approach, local optimization, pareto optimization approach, linear optimization. The techniques which comes under the approximation approach are genetic algorithm and particle swarm algorithm. The techniques which comes under the local optimization approach are dynamic programming, learning depth first search, simple additive weighing. The techniques which comes under the pareto optimization approach are genetic algorithm, particle swarm algorithm, multi-objective simple additive weighing. The techniques which comes under linear optimization approach is linear programming. The adaptive approach is sub-categorized into internal adaptation approach and external adaptation approach. The techniques which comes under the internal adaptation approach are AI-planning, reinforced learning, graph based approach, heuristic algorithm. The techniques which comes under external adaptation approach are protocol based approach, social network analysis approach.

The different workflow structure of web service composition are sequence, flow, switch and while. The web services composition method is classified as heuristics and non heuristics.

1.1 Motivation

There are many approaches proposed for web services composition. Every approach focus on a certain constraint based on the user's needs. This paper focuses on the web services composition based on the requirements of the particular architecture. However it is important to note that there is no such approach which satisfies all the requirements of the business needs. So discussing all these unique approaches gives us an idea about web service composition.

1.2 Contribution

In this paper, we present a survey on web services composition approaches. Web service are playing a vital role in today's world. All the online communications and transactions are happening because of these web services. So we started discussing about the web services composition approaches. We have discussed clearly about

each approach in depth. To the best of our awareness, we admit that this is the first survey on web services composition approaches which were introduced in the recent years.

1.3 Organization

The rest of this work is organized as follows. Section II presents Web service composition approaches. Section III concludes the paper.

2. Web Service Composition Approaches

2.1 An Approach for Automatic Web Service Composition which uses AI Planners

⁹Proposed an approach for automatic web service composition which uses AI planners. AI planning is the most widely used technique for practical applications such as robots, testing, controlling space vehicles and automated code synthesis. To obtain a specified goal, it starts with an original state, then looks out for a chain of actions and at last an execution plan is set. In this approach for solving the web service composition problem, composing a sequence of web services is converted into finding a chain of actions. Automated planning technique uses two steps for web service composition dynamically. In the first step, WSC problem is taken as input and translated into WSC planning problem. In the second step, for finding a composition plan automated planners are applied. Figure 1 represents the translation from a WSC problem to a WSC planning problem. A Web service composition architecture is proposed for validating the efficiency and feasibility of the proposed WSC approach. There are four steps in the Web service composition. In the first step by using the WSC graphical user interface, service providers publish their Web services to WSC system. In the second step by using the PDDL domain translator, WSC system translates services into planning domain.

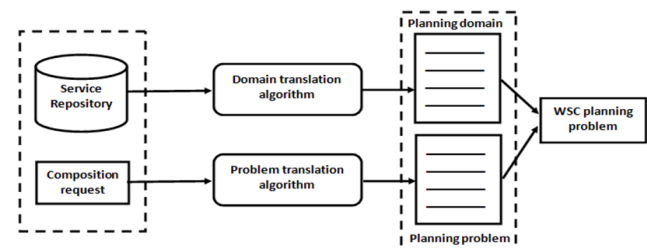


Figure 1. Data-End point level-0.

In the third step by using the PDDL problem translator, composition request is translated into planning problem. AI planner is invoked by WSC system for finding a composition plan. In the final step, a solution is returned to the requestor by the WSC system. Figure 2 represents the system architecture of Web service composition using efficient automated planners.

2.2 An Integrated Approach for Semantic Web Service Composition Automated through AI Planning

¹⁰Proposed an integrated approach for semantic web service composition which is automated through AI planning. The web service composition problem is transformed into a planning problem. For solving the planning problem AI

planning technique is used. PORSCE II, VLEPPO is used for implementing the approach. The overall approach consists of the steps such as Problem transformation, Visual representation (optional), Semantic enhancement, Solving through external planners, Composite service accuracy assessment, Service replacement (original) and Reverse transformation. Figure 3 represents the proposed approach steps.

2.2.1 PORSCE II

PORSCE II consists of the steps namely Semantic analysis, Problem transformation, Semantic awareness and relaxation, Solution acquisition and visualization, Composite service accuracy assessment, Atomic service unavailability handling and solution transformation.

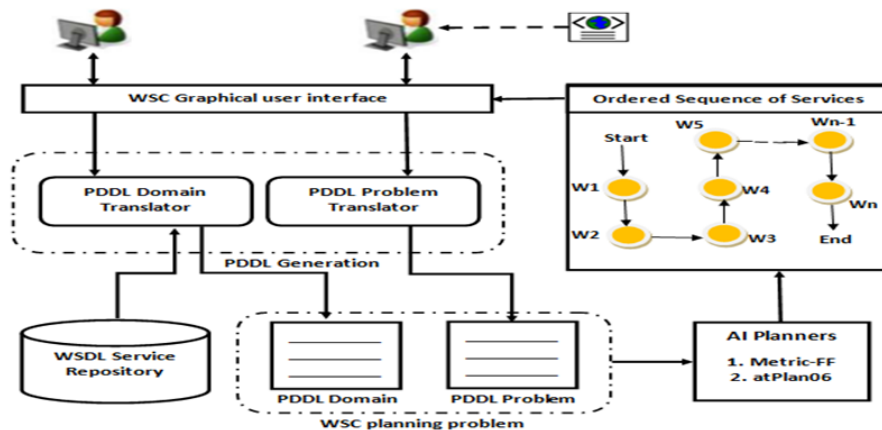


Figure 2. The system architecture of Web service composition using efficient automated planners.

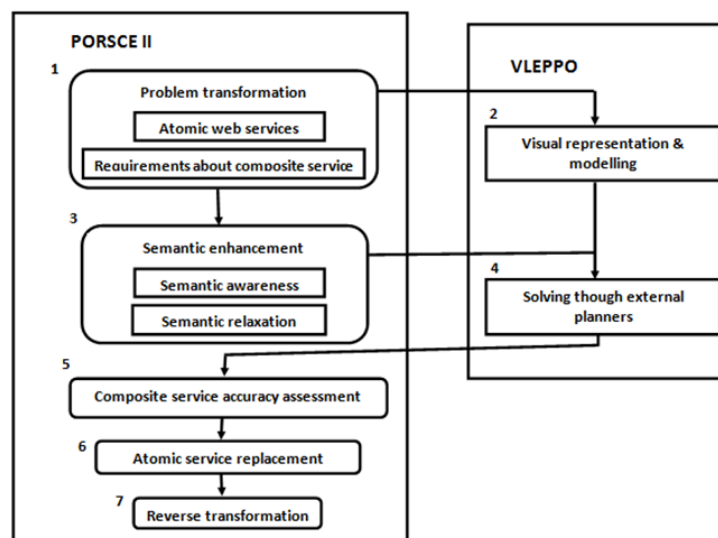


Figure 3. Proposed Approach Steps.

2.2.2 Semantic Analysis

OWL Ontology Manager (OOM) utilizes the semantic information of web services for enhancing the composition procedure. Here two ontology concepts are considered such as Hierarchical relationships and Semantic distance. Hierarchical relationships consists of four possible relationships. They are exact, plugin, subsume and sibling. Semantic distance consists of two methods¹¹ such as Edge-Counting Distance and Upward Copic Distance. Edge-Counting Distance describes about the number of edges on the shortest path between two concepts. Upward Copic Distance represents the set of superclasses of a concept including the concept itself.

2.2.3 Problem Transformation

The first step in problem transformation is translating the problems into planning terms. Based on the STRIPS (Stanford Research Institute Planning System) notation¹², planning problem is modeled. STRIPS notation is given as a tuple $\langle I; A; G \rangle$ where I represents the initial state, A represents a set of available actions, and G represents a set of goals.

2.3 Semantic Awareness and Relaxation

Semantic awareness facilitates planners to go with preconditions and effects appropriately, even if the terms used to refer to them in the web service OWL-S profiles differ¹³. Based on the semantic similarities semantic relaxation is done.

2.3.1 Solution Acquisition and Visualization

Based on the previous steps the created planning domain and problem is encoded into PDDL. In VLEPPO it is solved by external planners. In PORSCE II the derived plans which are structured in levels and may be sequential or parallel partially are visualized.

2.3.2 Composite Service Accuracy Assessment

Based on the usage of different planners and semantic relaxation, the users are offered multiple composite services. All these composite services may cover the demanded functionality of the user, but only some of such composite services are preferable. A statistics should be taken for these composite services. The statistics of distance quality metric, number of actions, accuracy metric and number of levels in a plan should be calculated.

2.3.3 Atomic Service Unavailability Handling

When a user tries to access a service and it becomes unsuccessful even though that service is accessed using an interface then atomic web service unavailability takes place. In the composite service description replacement of such unavailable service is necessary. Sometimes service replacement also deals with some other scenarios such user's unwillingness to a web service because of certain factors namely cost or time constraints, lack of trust and security concerns. Service replacement component discovers the alternate and the replacement atomic services which can be used as indicated. OOM advises the discovering of alternate atomic services. An action X may be taken as an alternative for an action Y of the plan when it does not disturb the plan sequence and the intermediate states.

2.3.4 Solution Transformation

The transformation of the produced composite service into OWL-S completes the composition process. OWL-S and domain ontologies are taken into consideration in this process. Composite processes are defined as a set of atomic processes by a framework which is established by OWL-S.

2.4 VLEPPO

VLEPPO consists of following steps: Visual Representation and Design, Interoperability, and Solving.

2.4.1 Visual Representation and Design

Visual component is simple and convenience. Using the graphical notations, planning domains and problems are represented. The plans can also be visualized which fulfills the PDDL+ standard representing composite services¹⁴. Visual representation and design consists of following steps: The Domain Entities and Relationships, Representing Operators, Representing Problems, and Syntax and Validity Checking.

2.4.2 Interoperability

The main characteristic of VLEPPO is the ability of importing from and exporting to PDDL. PDDL to serve in the comprehensive, manipulative and maintenance purposes, imports and visualizes planning domains and problems. Load/save functions are an alternate which can be used instead of import/export.

2.4.3 Solving

Solving occurs in two ways such as Interface with planners implemented as Web services and Solving planning problems locally. In the first way the process of discovering and communicating of web services is mentioned. The use of various implementation algorithms is also done. In the second way LPG-td¹⁵ and JPlan¹⁶ are used. LPG-td is proved as performing well by International Planning Competitions. JPlan is an open-source implementation of Graph plan.

2.5 Pre-computing Solutions for web service composition in a RDBMS (PSR)

¹⁷Proposes a system named Pre-computing Solutions for web service composition in a RDBMS (PSR) which uses relational database for searching the scalable and efficient web service composition.

2.5.1 PSR System

In the PSR system Web services composition problem is first formulated. Then an architectural overview of

the system is provided and its core components are also discussed.

2.5.1.1 Problem Definition

The problem is formulated based on the UDDI registry which consists of a set Web services and user query. The task is to find the another subset which belongs to a set of Web services. This should match the query including the composed Web services. A composition graph $G = (V, E)$ is defined. V represents a set of web services and E represents a set of edges.

3. Architectural Overview

Figure 4 represents the PSR architecture. The dotted box represents the core engine of the system. Web services are registered in UDDI registry by service providers. As soon as Web service gets added in the UDDI registry, WSDL description gets parsed and Web service information is extracted. A table in a relational database stores the input, output, and operation of the web services.

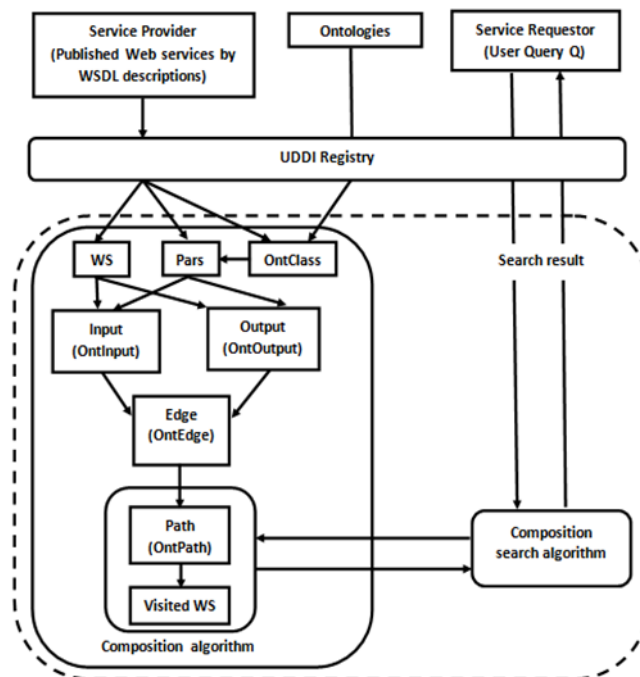


Figure 4. PSR Architecture Overview.

3.1 WSC Search Algorithm

3.1.1 Web Services and its Parameter

The table stores the input, output, and operation of the web services as tuples which are extracted from the WSDL file.

In the ws table, the web service id and name are stored. In the pars table, parameters names are stored. The parameters may be input and output parameters. So the input and output table is used here. The input table consists of input id, pars id and web service id. The output table consists of output id, pars id and web service id. Figure 5 shows all the above mentioned table.

WS table		Pars table			Input table			Output table		
Ws_id	name	I_id	P_id	WS_id	I_id	P_id	WS_id	O_id	P_id	WS_id
1	Ws1	1	1	1	1	1	1	1	3	1
2	Ws2	2	2	2	2	2	2	2	4	2
3	Ws3	3	3	3	3	3	3	3	5	3
4	Ws4	4	4	3	4	4	3	4	6	3
5	Ws5	5	5	4	5	5	4	5	7	3
6	Ws6	6	6	5	6	6	5	6	9	4
		7	7	6	7	7	6	7	10	4
		8	8	6	8	8	6	8	11	5
		9	9	5	9	9	5	9	11	6

Figure 5. Tables for storing Web Services.

3.2 Composition Graph

The Edge table stores the links of composition graph. Figure 6(a) represents the Edge Table. The column ws_s represents start vertex and ws_e represents end vertex. Figure 6(b) represents the composition graph representation. It represents the weighted, directed graph of the web services.

EdgeID	WS_s	WS_e
1	ws1	ws3
2	ws2	ws3
3	ws3	ws4
4	ws3	ws5
5	ws3	ws6
6	ws4	ws5

Figure 6. (a) Edge Table.

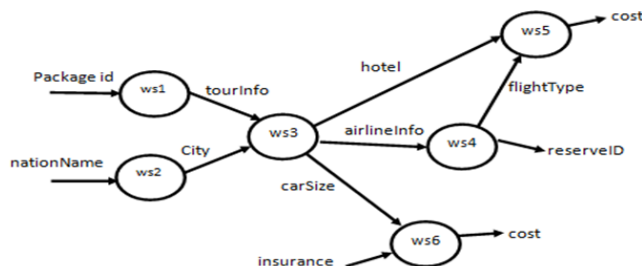


Figure 6. (b) Composition graph representation.

3.3 Web Service Composition

The Path table stores the PathID, web service start vertex (ws_s) and web service end vertex (ws_e). Figure 7(a) represents Path table. The VisitedWS table stores PathID, web service id (ws_id) and sequence (seq). Figure 7(b) represents the VisitedWS table.

Path table			VisitedWS table		
PathID	ws_s	ws_e	PathID	ws_id	seq
P1	ws1	ws3	P1	ws1	1
P2	ws2	ws3	P2	ws2	1
P3	ws3	ws5	P3	ws3	1
P4	ws1	ws5	P4	ws1	1
P5	ws2	ws5	P5	ws2	1
P6	ws3	ws4	P6	ws3	1
P7	ws1	ws4	P7	ws1	1
P8	ws2	ws4	P8	ws2	1
P9	ws3	ws6	P9	ws3	1
P10	ws1	ws6	P10	ws1	1
P11	ws2	ws6	P11	ws2	1
P12	ws4	ws5	P12	ws4	1
P13	ws3	ws5	P13	ws3	1
P14	ws1	ws5	P14	ws1	1
P15	ws2	ws5	P15	ws2	1

(a)

(b)

Figure 7. (a) Path Table. (b) VisitedWS table.

3.4 Pre-Computing Web Services Composition

It describes the pre-computing of WSC. Pre-computing is done by taking values from the edge table and storing them in Path and VisitedWS table. Two algorithms namely a bulk loading EP-Join algorithm (BEP-Join) and an incremental EP-Join algorithm (IEP-Join). In BEP-Join, a database is populated from the scratch. In IEP-Join, new web services are inserted into already existing database.

3.5 A Method for Automatic Web Service Composition Which Supports the Horn Clauses and Petrinets

¹⁸Proposed a method for automatic web service composition which supports the horn clauses and petrinets. The components of service composition framework are Service registry, rule builder, logical reasoner, Petrinet translator, Composition solver. Figure 8 represents the main framework of the service composition system.

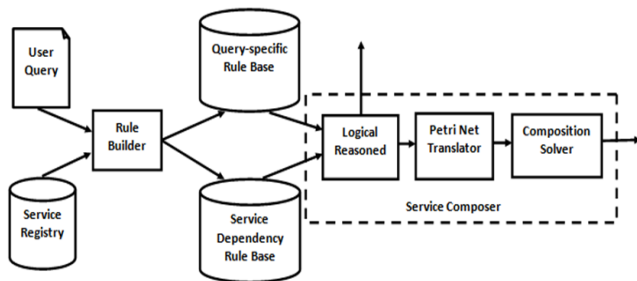


Figure 8. The main framework of the service composition system.

3.5.1 Service Registry

Service registry stores the semantic description of the registered web services.

3.5.2 Rule Builder

Rule builder generates web service dependency rules and a client query. The implementation is done based on the hypergraph theory.

A web service is defined as $ws = \{I, O, BC, QoS\}$ where
 I- input parameters of ws (set of semantic concepts),
 O- output parameters of ws (set of semantic concepts),
 BC- conditions imposed by the service provider on ws (set of behaviorial constraints),

QoS- quality attributes of ws (cost, response time, reliability and availability)

Here SAWSDL is chosen as the web service specification language. The behaviorial constraints can also be described in SAWSDL using model Reference. SAWSDL doesn't supports QoS characteristics. SLA (Service Level Agreement) can be used by service providers for describing QoS attributes.

3.5.3 Logical Reasoner

Logical reasoner determines the existence of composite service that satisfies the client query. For propositional logic, logical reasoner is implemented based on the forward-chaining algorithm.

3.5.4 Petri Net Translator

Petri net translator obtains the petri net representation of the selected clause set.

3.5.5 Composition Solver

Using the structural analysis techniques composition

solver generates the composite services. This method concentrates on the behavior constraints but generating behavior graph very expensive in this method. The main purpose of choosing the petrinets is to model the rule set and its structural analysis technique. Semantic Annotation for WSDL and XML Schema (SAWSDL) is used for the implementation.

3.6 An approach for Analyzing the QoS of Web Service Composition which has Complex Structures

¹⁹Proposed an approach for analyzing the QoS of web service composition which has complex structures. Sequential, parallel, loop and conditional patterns are patterns for composite web services. For computing the QoS of web service composition service graph is used. QoS explorer is used for implementing the web service composition which has complex structure. This approach provides complete information of QoS even though there exists composition structures such as MEME loop patterns and unstructured conditional patterns. In this approach it is assumed that each task has a fixed QoS and based on that the QoS of web service composition is calculated.

3.7 A Novel Architecture for Composing Web Services

²⁰Proposed a novel architecture for composing web services. This architecture has three components namely a business model, a description framework and a composition framework.

3.7.1 Business Model

Figure 9 represents the extended web services business model. The different types interactions carried on is explained here. 1) Using the put interaction, the primitive web services are published by the third party. 2) Using the register interaction, the composers registers for obtaining the notifications of the services the requested the requestors. 3) Using the find interaction, the requester finds whether the service exists in the registry or not 4) Inform interaction notifies the composer about the composition if possible. 5) Using the get interaction for obtaining the composition request, the composer communicates with the requestor. 6) The requestor uses give interaction for specifying the service request. 7)

For creating the requested services the composer uses its techniques, the location interaction is used here. 8) Using the publish interaction, the created composite service is published in the registry. 9) Bind interaction provides the composed service to the requester. 10) Web service composer binds to the third party service provider.

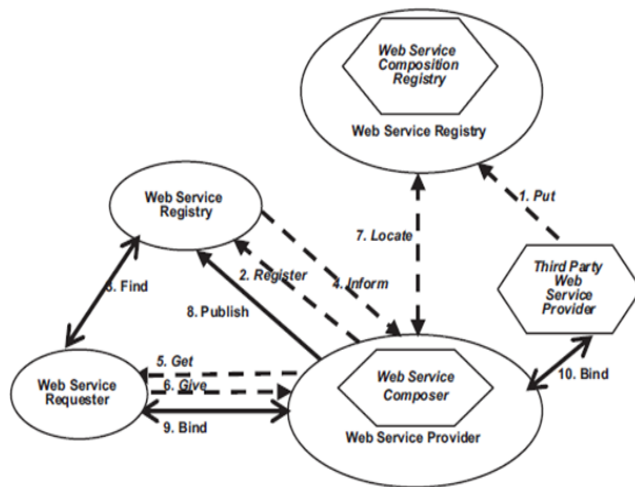


Figure 9. Tables for storing Web Services.

3.7.2 Description Framework

The Description framework describes the four characteristics of the web services. For semantic and functional description SAWSDL²¹ is used. Figure 10 shows the SAWSDL description of the dating service.

For behavioral description MSC is used and for non-functional description NFSL is used. Non-functional Specification Language (NFSL) is an xml based language which describes the non-functional parameters is also proposed here. Figure 11 and Figure 12 shows the MSC and NFSL description of the dating service.

The integration of these three languages is done using HOL²³. Integration plays a vital role since it manipulates the four characteristics of the web services. This allows matching, creation and assembly simple in a domain.

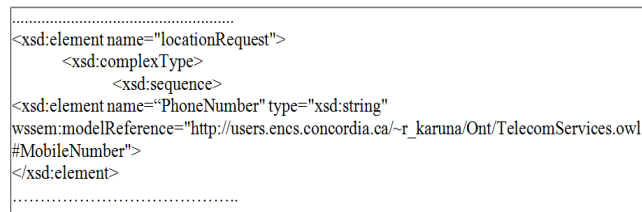


Figure 10. Part of the SAWSDL description of the dating service.

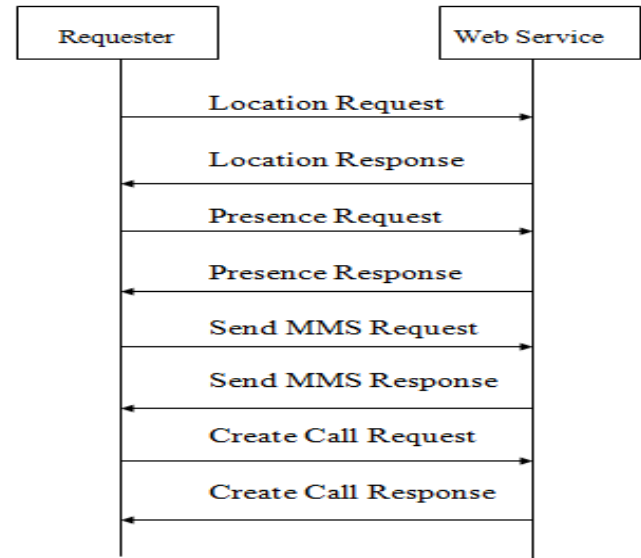


Figure 11. MSC description of the dating service.

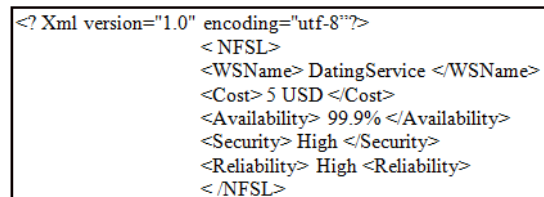


Figure 12. NFSL description of the dating service.

3.7.3 Composition Framework

The composition framework functionally consists of four modules namely communication, request processing, composition, and execution-time adaptation. Figure 13 shows the composition framework. In the figure, numbers 1 and 2 represents the framework's external communication with the registry. The register and inform interactions are used here. The numbers 3 and 4 represents the framework's external communication with the requester. The get and give interactions are used here. The number represents the framework's external communication with the composition registry. The locate interaction is used here.

The three techniques used here are namely Matchmaking technique, Categorization technique and Assembly technique.

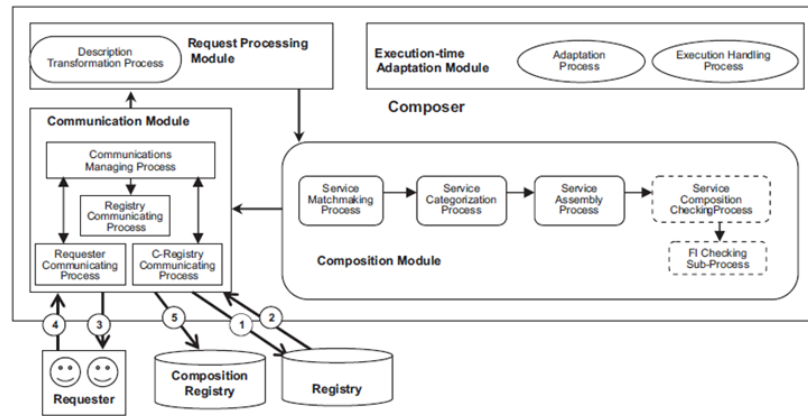


Figure 13. The composition framework.

3.7.4 Matchmaking Technique

At a high order logic domain, the matchmaking technique manipulates Web services using the HOL²⁴. The basic principle of matchmaking technique is to find whether a third-party service matches the requested service. If it matches, it is directly provided to the requester. If it doesn't matches, using partially-matched service a composition is carried out.

3.8 Categorization Technique

Categorization technique categorizes the partially-matched services. During matching different types of partially-matched are discovered. So this technique is required²².

3.9 Assembly Technique

Assembly technique selects the categorized services. It selects them from different categories then for orchestrating it uses BPEL. At last it selects a composed service which is best-assembled.

4. Conclusion

In this paper, we presented a survey on web services composition approaches. We have surveyed the latest six approaches, discussed it's advantages, disadvantages and simulation tools used for implementing. The outcome of this survey explored that there are many approaches for web service composition which is specific based on user and business needs. We have also observed that all the approaches have not concentrated on the major QoS factors.

5. References

1. Batouche B, Naudet Y, Guinand F. Semantic Web Services Composition Optimized by Multi-objective Evolutionary Algorithms. The Proceeding of 5th International Conference on Internet and Web Applications and Services (ICIW), Barcelona. 2010. p. 180–85.
2. Ghafarian T, Kahani M. Semantic web service composition based on ant colony optimization method. The Proceedings of 1st International Conference on Networked Digital Technologies, NDT'09, Ostrava. 2009. p. 171–76.
3. Amiri MA, Serajzadeh H. QoS aware web service composition based on genetic algorithm. The Proceedings of 5th International Symposium on Telecommunications (IST), Tehran. 2010; 502–7.
4. Amiri MA, Serajzadeh H. Effective web service composition using particle swarm optimization algorithm. The Proceedings of 5th International Symposium on Telecommunications (IST), Tehran. 2012; 1190–94.
5. Liu M, Wang M, Shen W, Luo N, Yan J. A quality of service (QoS)-aware execution plan selection approach for a service composition process. Future Generation Computer Systems. 2012; 28(7):1089–89.
6. Lu Y, Xie Q, Kliot G, Geller A, Larus JR, Greenberg A. Join-Idle-Queue: A novel load balancing algorithm for dynamically scalable web services. Performance Evaluation. 2011; 68(11):1056–71.
7. Amudhavel J, et al. An robust recursive ant colony optimization strategy in VANET for accident avoidance (RACO-VANET). International Conference on Circuit, Power and Computing Technologies (ICCPCT); Nagercoil. 2015. p. 1–6.
8. Amudhavel J, et al. A krill herd optimization based fault tolerance strategy in MANETs for dynamic mobility. International Conference on Circuit, Power and Computing Technologies (ICCPCT); Nagercoil. 2015. p. 1–7.
9. Amudhavel J, Prabu U, Dhavachelvan P, Moganarangan N, Ravishankar V, Baskaran R. Non-homogeneous hidden Markov model approach for load balancing in web server

- farms (NH2M2-WSF). Global Conference on Communication Technologies; Thuckalay. 2015. p. 843–5.
10. Hatzi O, Vrakas D, Nikolaidou M, Bassiliades N. An integrated approach to automated semantic web service composition through planning. *IEEE Transactions on Services Computing*. 2012; 5(3):319–32.
 11. Hatzi O, Meditskos G, Vrakas D, Bassiliades N, Anagnos D, Vlahavas I. Semantic Web Service Composition Using Planning and Ontology Concept Relevance with PORSCE II. The Proceedings of IEEE/WIC/ACM Int'l Joint Conference on Web Intelligence and Intelligent Agent Technologies. 2009; 1:418–21.
 12. Fikes RE, Nilsson NJ. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. The Proceedings of Second Int'l Joint Conf Artificial Intelligence. 1971; 2(3):189–208.
 13. Paolucci M, Kawamura T, Payne T, Sycara K. Semantic Matching of Web Services Capabilities. The Proceedings of 1st International Semantic Web Conference, Italy. 2002. p. 333–47.
 14. Fox M, Long D. PDDL+: Modeling Continuous Time Dependent Effects. The Proceedings of 3rd Int'l NASA Workshop Planning and Scheduling for Space. 2002; 1–10.
 15. Gerevini A, Saetti A, Serina I. LPG-td: A Fully Automated Planner for PDDL2.2 Domains. The Proceeding of 14th Int'l Conf Automated Planning and Scheduling (ICAPS). 2004. p. 1–2.
 16. JPlan. Available from: <http://sourceforge.net/projects/jplan>, 2012. Date accessed 25/4/2014.
 17. Lee D, Kwon J, Lee S, Park S, Hong B. Scalable and efficient web service composition based on a relational database. *Journal of systems and software*. 2011; 84(12):2139–55.
 18. Tang X, Jiang C, Zhou M. Automatic Web service composition based on Hornclauses and Petrinets. *Journal on Expert Systems with Applications*. 2011; 38(10):13024–31.
 19. Hzung H, Zhao W, Yang J, Bouguettaya A. QoS Analysis of Web service composition with complex structures. *IEEE Transactions on Services Computing*. 2012; 6(3):373–86.
 20. Karunamurthy R, Khendek F, Glitho RH. A novel architecture for Web Service Composition. *Journal of Network and Computer Application*. 2012; 35(2):787–802.
 21. Web Service Description Language. Available from: <http://www.w3c.org/TR/sawSDL/>. 28/08/2007.
 22. Karimi M, Esfahani FS, Noorafza N. Improving Response Time of Web Service Composition base on QoS Properties. *Indian Journal of Science and Technology*. 2015; 8(16):1–8.
 23. Manickasankari N, Arivazhagan D, Vennila G. A Survey on Query Processing in Mobile Database. *Indian Journal of Science and Technology*. 2014 Oct; 7(S6):32–4. Doi:10.17485/ijst/2014/v7iS6/54590
 24. Ashwin JS, Praveen JS, Manoharan N. Optimization of SRAM array structure for energy efficiency improvement in advanced CMOS technology. *Indian Journal of Science and Technology*. 2014 Oct; 7(S6):35–9. Doi: 10.17485/ijst/2014/v7iS6/54581.